IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

| | |
|---|---|
| In re Application of: | Michael E. Brown et al. |
| Serial No.: | 09/961,218 |
| Date Filed: | September 21, 2001 |
| Group Art Unit: | 2145 |
| Confirmation No.: | 4097 |
| Examiner: | Bhatia, Ajay M. |
| Title: | **SYSTEM AND METHOD FOR NAMING HOSTS IN A DISTRIBUTED DATA PROCESSING SYSTEM** |

**MAIL STOP – APPEAL BRIEF - PATENTS**
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

## APPEAL BRIEF

Further to the Notice of Appeal submitted on March 5, 2007, and the Notice of Panel Decision from Pre-Appeal Brief Review mailed September 13, 2007, Appellants hereby submit this Appeal Brief pursuant to 37 C.F.R. § 41.37.

## APPELLANTS' BRIEF (37 C.F.R. § 41.37)

This brief is submitted in support of Appellants' Notice of Appeal from the Final Office Action mailed October 4, 2006 and Advisory Action mailed February 14, 2007, finally rejecting Claims 1-12 and 14-22 of the subject application.

## I. REAL PARTY IN INTEREST

This application is currently owned by Dell Products L.P., as indicated by an assignment recorded on November 31, 2001, in the Assignment Records of the United States Patent and Trademark Office at Reel 012197, Frame 0376.

## II. RELATED APPEALS AND INTERFERENCES

There are no known appeals or interferences which will directly affect or be directly affected by or have a bearing on the Board's decision regarding this appeal.

## III. STATUS OF CLAIMS

Claims 1-12 and 14-22 are pending in this application and all stand rejected under a Final Office Action mailed October 4, 2006. Appellants present Claims 1-12 and 14-22 for appeal. Appendix A shows all pending claims.

## IV. STATUS OF AMENDMENTS

No amendments have been filed subsequent to final rejection.

## V. SUMMARY OF CLAIMED SUBJECT MATTER

*Independent Claim 1* relates to a method for automatically naming hosts in a distributed data processing system. (*See*, *e.g.*, Specification Page 8, lines 2-4; Page 10, lines 13-14; Page 13, lines 9-18; FIG. 4 flowchart). A unique identifier (UID) is received at a cluster controller from each of a plurality of hosts in communication with the cluster controller, while at least one of the plurality of hosts is executing in a pre boot execution environment. (*See*, *e.g.*, Specification Page 11, lines 8-17; Page 14, lines 4-9; FIG. 4, step

204). In response to receiving the UIDs, the plurality of hosts are caused to produce ready signals. (*See*, *e.g.*, Specification Page 11, lines 24-29; FIG. 4 step 212). User input is received from a first host among the plurality of hosts, the user input comprising notification of the insertion of a disk within the first host. (*See*, *e.g.*, Specification Page 12, lines 3-11; FIG. 4 step 216). In response to receiving the user input from a first host, a first host name is associated with the UID for the first host. (*See*, *e.g.*, Specification Page 12, lines 12-26; FIG. 4 steps 218-224). After associating the first host name with the UID for the first host, the first host is caused to produce a completion signal. (*See*, *e.g.*, Specification Page 12, line 26 to Page 13, line 3; FIG. 4 step 228). User input is subsequently received from a second host among the plurality of hosts. (*See*, *e.g.*, Specification Page 13, lines 4-8; FIG. 4 steps 230 and 216). The operations of receiving replies from hosts, associating host names with UIDs, and causing hosts to produce completion signals are then repeated, until each of the plurality of hosts has been named, such that the user input dictates the order in which host names are assigned to the multiple hosts. (*See*, *e.g.*, Specification Page 13, lines 4-8; FIG. 4 steps 230 and 214-228).

*Independent Claim 9* relates to a program product for automatically naming hosts in a distributed data processing system. (*See*, *e.g.*, Specification Page 8, lines 11-25; Page 9, line 25 to Page 10, line 2; FIG. 1 cluster controller 20). The program product includes computer instructions enabling a controller to receive a unique identifier (UID) from a first host in communication with a cluster controller, at least one of the plurality of hosts not having a fully functional operating system present thereon. (*See*, *e.g.*, Specification Page 8, lines 11-25; Page 11, lines 8-17; Page 14, lines 4-9; Page 16, lines 6-13; FIG. 4, step 204). The computer instructions enable the controller to cause the first host to produce a ready signal in response to receiving the UID. (*See*, *e.g.*, Specification Page 8, lines 11-25; Page 11, lines 24-29; FIG. 4 step 212). The computer instructions also enable the controller to receive user input from the first host, the user input. (*See*, *e.g.*, Specification Page 8, lines 11-25; Page 12, lines 3-11; FIG. 4 step 216). The computer instructions also enable the controller to associate a first host name with the UID for the first host in response to receiving the user input from the first host. (*See*, *e.g.*, Specification Page 8, lines 11-25; Page 12, lines 12-26; FIG. 4 steps 218-224). The computer instructions also enable the controller cause the first host to produce

a completion signal after associating the first host name with the UID for the first host. (*See*, *e.g.*, Specification Page 8, lines 11-25; Page 12, line 26 to Page 13, line 3; FIG. 4 step 228).

*Independent Claim 16* relates to a data processing system for automatically naming hosts in a distributed data processing system. (*See*, *e.g.*, Specification Page 8, lines 2-4; FIG. 1 system 10). The data processing system comprises a network interface in communication with a plurality of hosts, a processor in communication with the network interface, data storage in communication with the processor, and computer instructions stored in the data storage. (*See*, *e.g.*, Specification Page 8, lines 11-29; FIG. 1 cluster controller 20 and hosts 22). The computer instructions are executable to receive a unique identifier (UID) from each of a plurality of the plurality of hosts. (*See*, *e.g.*, Specification Page 8, lines 11-25; Page 11, lines 8-17; Page 14, lines 4-9; Page 16, lines 6-13; FIG. 4, step 204). The computer instructions are further executable to cause the plurality of hosts to produce ready signals in response to receiving the UIDs. (*See*, *e.g.*, Specification Page 8, lines 11-25; Page 11, lines 24-29; FIG. 4 step 212). The computer instructions are further executable to receive user input from a first host among the multiple hosts, the user input comprising a signal indicative of an insertion of a disk within a disk drive of the first host. (*See*, *e.g.*, Specification Page 8, lines 11-25; Page 12, lines 3-11; FIG. 4 step 216). The computer instructions are further executable to associate a first host name with the UID for the first host without regard to data, if any, stored on the disk in response to receiving the user input from the first host. (*See*, *e.g.*, Specification Page 8, lines 11-25; Page 12, lines 12-26; FIG. 4 steps 218-224). The computer instructions are further executable to cause the first host to produce a completion signal after associating the first host name with the UID for the first host. (*See*, *e.g.*, Specification Page 8, lines 11-25; Page 12, line 26 to Page 13, line 3; FIG. 4 step 228). The computer instructions are further executable to receive user input from a second host among the plurality of hosts. (*See*, *e.g.*, Specification Page 8, lines 11-25; Page 13, lines 4-8; FIG. 4 steps 230 and 216). The computer instructions are further executable to repeat the operations of receiving replies from hosts, associating host names with UIDs, and causing hosts to produce completion signals, until each of the plurality of hosts has been named, such that the user input dictates the order in which host names are assigned to the plurality of hosts. (*See*, *e.g.*, Specification Page 8, lines 11-25; Page 13, lines 4-8; FIG. 4 steps 230 and 214-228).

## VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Appellants request that the Board review the following grounds of rejection:

(1) Whether Claims 1-4, 6, 16-18, and 20[1] are allowable under 35 U.S.C. § 103(a) over U.S. Patent Application Publication 2002/0161868 issued to Chakkalamattam J. Paul et al. ("*Paul*"), in view of U.S. Patent 5,974,547 issued to Yevgeniy Klimenko ("*Klimenko*");

(2) Whether Claims 9-11 and 22 are allowable under 35 U.S.C. § 103(a) over *Paul* in view of *Klimenko*; and

(3) Whether Claims 5, 7, 8, 12, 14, 15, 19, and 21 are allowable under 35 U.S.C. § 103(a) over *Paul* in view of *Klimenko*, and further in view of U.S. Patent No. 5,864,656 issued to Jee-Kyoung Park ("*Park*").

## VII. ARGUMENT

### *A. Rejection under 35 U.S.C. § 103(a) over* Paul *in view of* Klimenko.

#### (1) Claims 1-4, 6, 16-18, and 20

*Summary:*

*The rejection of independent Claims 1 and 16 as being unpatentable under 35 U.S.C. § 103(a) over* Paul *in view of* Klimenko *is improper because neither of* Paul *nor* Klimenko, *individually or in combination, disclose, teach or suggest the combination of elements recited in Claims 1 or 16.*

*The rejection of dependent Claims 2-4 and 6 is improper at least because they depend from and provide further patentable elements to independent Claim 1.*

*The rejection of dependent Claims 17-18 is improper at least because they depend from and provide further patentable elements to independent Claim 16 .*

It is a bedrock principle of patent law that, in order to establish a prima facie case of obviousness, the references cited by an Examiner in an Office Action must disclose all claimed elements. *In re Royka*, 490 F.2d 981, 180 U.S.P.Q. 580 (C.C.P.A. 1974). Furthermore, according to § 2143 of the Manual of Patent Examining Procedure, to establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally

[1] Claim 20 was not specifically addressed in the Final Office Action. However, based on the rejection of similar Claim 6, Appellants assume that the Examiner intended to reject Claim 20 under *Paul* in view of *Klimenko*, and consider Claim 20 so rejected for the purposes of this Appeal

available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim elements. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, not in applicant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 U.S.P.Q.2d 1438 (Fed. Cir. 1991).

Appellants contend that the art cited by the Examiner cannot render the rejected claims obvious, at least because the cited prior art references, taken separately or in combination, fail to disclose, teach or suggest all elements of the rejected claims.

For example, Claim 1 recites:

> 1. A method for automatically naming hosts in a distributed data processing system, the method comprising:
>
> (a) receiving a unique identifier (UID) at a cluster controller from each of a plurality of hosts in communication with the cluster controller, while at least one of the plurality of hosts is executing in a pre boot execution environment;
>
> (b) in response to receiving the UIDs, causing the plurality of hosts to produce ready signals;
>
> (c) receiving user input from a first host among the plurality of hosts, the user input comprising notification of the insertion of a disk within the first host;
>
> (d) in response to receiving the user input from a first host, associating a first host name with the UID for the first host;
>
> (e) after associating the first host name with the UID for the first host, causing the first host to produce a completion signal;
>
> (f) receiving user input from a second host among the plurality of hosts; and
>
> (g) repeating the operations of receiving replies from hosts, associating host names with UIDs, and causing hosts to produce completion signals, until each of the plurality of hosts has been named, such that the user input dictates the order in which host names are assigned to the multiple hosts.

[Note: reference letters (a)-(g) are included for reference only and are not actually included in the pending claims.]

Appellants maintain that *Paul* and *Klimenko*, whether considered alone or in combination, fail to teach or suggest nearly every element -- in particular, elements (a), (b), (c), (d), (e), and (g) -- of Claim 1. As Appellants show below, the passages of *Paul* and *Klimenko* cited by the Examiner clearly do not teach these or suggest elements. In fact, in some instances, the portions of *Paul* and *Klimenko* cited by the Examiner are not even remotely similar to the elements of Claim 1.

Appellants provide below an element-by-element explanation of how *Paul* or *Klimenko*, whether considered alone or in combination, fail to teach or suggest each of these elements. Appellants provided a similar detailed explanation in the Response to Final Office Action dated March 5, 2007, and requested that the Examiner explain how certain elements could possibly be found in *Paul* or *Klimenko*. However, the Examiner declined to provide an explanation. Instead, the Examiner presumptively accused Appellants' representative of knowing the claims to be unpatentable, yet intentionally electing to read the prior art out of context:

> It appears from reading applicant's representative's remarks that he is well aware of the teachings of the prior art and that the prior art teaches the features disclosed in the claims, but has elected to read only the portion of the reference cite[d] with out context. Therefore examiner suggest applicant's representative's read the entire prior cited and submit amendments to overcome the prior art to further prosecution of the case."

(Feb. 14, 2007 Advisory Action, Page 2).

In light of this accusation, Appellants realized that the Examiner did not intend to actually identify specific portions of *Paul* or *Klimenko* that disclosed the elements of Appellants' claims, and thus Appellants were left with the option of appealing the Examiner's decision.

Provided below is an element-by-element explanation of how each of elements (a), (b), (c), (d), (e), and (g) of Claim 1 are not taught or suggested by *Paul* or *Klimenko*.

**(a) receiving a unique identifier (UID) at a cluster controller from each of a plurality of hosts in communication with the cluster controller, while at least one of the plurality of hosts is executing in a pre boot execution environment;**

The Examiner alleged in the Final Office Action mailed October 4, 2006 ("Final Office Action") that this element is disclosed at *Paul*, paragraph 32. (Final Office Action, page 3). However, paragraph 32 of *Paul* simply discloses:

> **[0032]** PXE specifies the protocols by which a client requests and downloads an executable image from a boot server. PXE does not specify the operational details and functionality of the network bootstrap program (NBP) that the client receives from the server, i.e. the remote boot image downloaded by the PXE client via TFTP or MTFTP (Multicast TFTP). In general, the execution of the downloaded NBP initiates a series of processing steps on the client that ultimately will result in the client being ready for use by its user. Typically, the NBP will use an application program interface (API) specified by PXE and provided by the client PXE support to request and install additional files via M/TFTP from the boot server containing executable images of an operating system, appropriate communications and other device drivers, and other system software. The NBP will then transfer client execution to the operating system which can use either PXE or its own communications support to request user-specific configuration information and application software executable images from the boot server for installation on the client.

As Appellants explained in the Response to Final Office Action dated February 5, 2007 ("Final Office Action Response"), this passage merely explains that the Preboot Execution Environment (PXE) specifies the communication protocols that a client may use to request a network bootstrap program (NBP) from a boot server, and that when executed, the NPB requests and installs additional files from the boot server in order to ready the client for use.

Paragraph 32 discloses nothing that could be equated with a "unique identifier (UID)," much less "receiving a unique identifier (UID) at a cluster controller from each of a plurality of hosts," as recited in Claim 1. After reviewing Paragraph 32 in light of element (a) of Claim 1, Appellants requested that the Examiner indicate precisely which element recited in Paragraph 32 the Examiner believes could be equated with the "unique identifier (UID)" that is received at a cluster controller from each of a plurality of hosts. (*See* Final

Office Action Response, page 9). However, the Examiner refused to provide such indication, instead accusing the Appellants of reading the passages "with out context" (*See* Advisory Action, Page 2), as discussed above.

> **(b) in response to receiving the UIDs, causing the plurality of hosts to produce ready signals;** (emphasis added)

The Examiner alleges that this element is disclosed at *Paul*, paragraph 35. (Final Office Action, page 3). However, paragraph 35 of *Paul* simply discloses:

> **[0035]** In the PXE protocol, DHCP option fields are used to perform the following: (a) distinguish between DHCPDISCOVER and DHCPREQUEST packets sent by a client as part of this extended protocol from other packets that the DHCP server or boot server might receive; (b) distinguish between DHCPOFFER and DHCPACK packets sent by a DHCP or Proxy DHCP server as part of this extended protocol from other packets that the client may receive; (c) convey the client system's ID (in most cases, the client's UUID--Universally Unique Identifier) to the DHCP and boot server; (d) convey the client system's architecture type to the DHCP server and boot server; and (e) convey the boot server type from which the client is requesting a response. Based on any or all of the client network adapter type, system architecture type, and client system ID, the boot server returns to the client the file name (on the server) of an appropriate NBP executable. The client downloads the specified NBP executable into memory and then executes it. As noted above, the functionality within the downloaded NBP is not specified by the PXE protocol.

The passage discusses various functions performed by DHCP option fields in the PXE protocol, which includes conveying various data between a client, a DHCP server, and a boot server, including "convey[ing] the client system's ID (in most cases, the client's UUID--Universally Unique Identifier) to the DHCP and boot server." Based on the client UUID and other data, the boot server sends the client the file name of a particular network bootstrap program (NBP) executable, which the client then downloads and executes in order to boot the client.

The passage does not disclose anything about "causing the plurality of hosts to produce ready signals," much less doing so "in response to receiving [unique identifiers from each of the plurality of hosts]," as recited in Claim 1. There is no disclosure of a ready

signal, much less causing multiple hosts/clients to produce a ready signal in response to receiving identifiers from the multiple hosts/clients.

Appellants requested that the Examiner indicate precisely which action disclosed in Paragraph 35 of *Paul* could be equated with "causing [a] plurality of hosts to produce ready signals," in response to receiving unique identifiers from each of the plurality of hosts. (*See* Final Office Action Response, page 9). However, the Examiner merely accused the Appellants of reading the passages "with out context," as discussed above.

**(c) receiving user input from a first host among the plurality of hosts, the user input comprising notification of the insertion of a disk within the first host;**

The Examiner correctly acknowledged in the Final Office Action that *Paul* fails to teach this element. (Final Office Action, Page 4). Instead, the Examiner alleges that this element is disclosed at *Klimenko*, Col. 4, lines 17-63. *Id.* However, Column 4, lines 17-63 of *Klimenko* merely recites:

> While the boot process is occurring but prior to the availability of any client O/S-based network support, client hard disk emulation occurs through appropriate calls made to an interrupt (Interrupt 13 or simply "Int 13") handler. Through such calls, appropriate sectors in the client image file are initially downloaded, via a real-mode network adapter (NIC) driver and the Int 13 Handler to remotely install various components of the O/S into client PC. The actual client hard disk emulation process is provided through a real mode procedure that executes as part of Int 13 Handler. In essence, the real mode procedure determines, based on values of status flags, whether the client O/S is then capable of handling a network request for sector access of the client image file. If the client O/S has not then progressed to that point in its boot process, the real mode procedure processes that request, in real mode, through the Int 13 Handler.
>
> As a client O/S kernel is installed and initialized during the boot process, the kernel installs and activates various device drivers, including the inventive LANHDVSD.VXD procedure. This procedure is compliant with both the Int 13 Handler and with the O/S, specifically, in the case of Windows 95 O/S, a network driver (NDIS--network driver interface specification) kernel therein and the O/S input/output subsystem (IOS). The inventive procedure, which executes as a protected mode driver, contains two asynchronous procedures. These asynchronous procedures, by setting and testing appropriate flags used as processing state semaphores, collectively control the transition of

> hard disk requests to the networked client image from the Int 13 Handler to the client O/S depending upon, as the client O/S is then booting, the O/S resources that are then available. During early phases of the boot process, insufficient O/S components have been loaded and activated to provide client O/S supported network access. Consequently, client hard disk access requests are handled through the Int 13 Handler. Whenever sufficient O/S resources become available to permit network access through the client O/S, the asynchronous procedures permit these requests to be serviced by the NDIS and IOS components of the client O/S, so as to provide O/S supported network access, rather than by the Int 13 Handler. Hence, these asynchronous procedures collectively assure, in conjunction with Int 13 Handler, seamless and continuous client hard disk emulation during the real-protected transitory state.

As Appellants explained in the Final Office Action Response, there is nothing in this passage regarding the insertion of a disk within a host, much less receiving a "notification of the insertion of a disk within [a] host." In fact, the only mention of a disk in the passage regards a "client hard disk emulation process," which does not concern the insertion of a disk within a host or a notification of such an insertion of a disk. Thus, *Klimenko* cannot teach this element.

> **(d) in response to receiving the user input from a first host, associating a first host name with the UID for the first host;** (emphasis added)

The Examiner alleges that this element is disclosed at *Paul*, paragraph 35. (Final Office Action, page 3). Paragraph 35 of *Paul* is reproduced above regarding element (b).

As Appellants explained in the Final Office Action Response, Paragraph 35 fails to disclose a client having *both* a "host name" *and* a "unique identifier (UID)" for a particular host, much less *associating* the host name with the "unique identifier (UID) for the particular host. Even assuming for the sake of argument that the "client's UUID--Universally Unique Identifier" disclosed in Paragraph 35 of *Paul* could be equated with the "unique identifier (UID) for [a] host" of Claim 1, *Paul* fails to disclose associating a host name with the client's UUID. Further, Paragraph 35 fails to disclose making any association of names or identifiers for a host "in response to receiving . . . user input from [the] host."

After reviewing Paragraph 35 of *Paul* in light of element (d) of Claim 1, Appellants requested that the Examiner indicate precisely which elements disclosed in Paragraph 35 can be equated with the "first host name" and the "unique identifier (UID)" for a first host, as well as the operation that can be equated with associating a "first host name" with the "unique identifier (UID)" for a first host. (*See* Final Office Action Response, page 10). However, the Examiner merely accused the Appellants of reading the passages "with out context," as discussed above.

**(e) after associating the first host name with the UID for the first host, causing the first host to produce a completion signal;**

The Examiner alleges that this element is disclosed simply at "(Paul, )." (Final Office Action, page 3). This is the full extent of the Examiner's explanation of how *Paul* teaches element (e) of Claim 1. Appellants pointed out this deficiency to the Examiner (*See* Final Office Action Response, page 10), but the Examiner declined to correct the deficiency.

Appellants assume that the Examiner either forgot to include the paragraph reference, or could not find any portion of *Paul* that could be equated with this element of Claim 1. In any event, the rejection is improper because the Final Office Action failed to cite the prior art with sufficient specificity under 35 U.S.C. § 132 and 37 C.F.R. § 1.104 to allow appellants to adequately respond to the rejections.

First, the Final Office Action does not comply with the intent and purpose of 35 U.S.C. § 132 because it fails to properly identify and clearly explain the portions of the cited prior art that allegedly teach element (e) of Claim 1.

In addition to defeating the intent and purpose of 35 U.S.C. § 132, the Final Office Action does not comply with the requirements of 37 C.F.R. § 1.104 due to lack of specificity. 37 C.F.R. § 1.104(c)(2) states:

> In rejecting claims for want of novelty or obviousness, the examiner must cite the best references at his or her command. When a reference is complex or shows or describes inventions other than that claimed by the applicant, the particular part relied on must be designated as nearly as practicable. The pertinence of each reference, if not apparent, must be *clearly explained* and each rejected claim specified. (emphasis added).

Because the Final Office Action fails to cite *any* particular portion of *Paul* that allegedly teaches the element of Claim 1 recited above, the Final Office Action fails to comply with both 35 U.S.C. § 132 and 37 C.F.R. § 1.104.

> **(g) repeating the operations of receiving replies from hosts, associating host names with UIDs, and causing hosts to produce completion signals, until each of the plurality of hosts has been named, such that the user input dictates the order in which host names are assigned to the multiple hosts.** (emphasis added)

The Examiner alleges that these elements are disclosed at *Paul*, paragraphs 52 and 64. (Final Office Action, pages 3-4). Paragraph 52 of *Paul* discloses that PXE clients broadcast initial request packets in a network with redundant boot servers. An initial request packet broadcast from a PXE client is received by all boot servers in the network. Any of the boot servers having a properly configured DHCP server service can respond to the initial broadcast request packet. The PXE client can receive one or more boot server responses and choose a response which directs it to a boot server to complete its remote boot.

Paragraph 64 of *Paul* discloses that a determination is made regarding the availability of a local server for booting additional clients and the availability of other servers that have reported that they are booting additional clients. A determination is then made as to whether or not a PXE Proxy service is currently executing.

Thus, these paragraphs fail to disclose "associating host names with UIDs" or "causing hosts to produce completion signals." These paragraphs also clearly fail to disclose repeating a set of operations "until each of the plurality of hosts has been named, such that the user input dictates the order in which host names are assigned to the multiple hosts." Nothing in Paragraphs 52 or 64 of *Paul* discloses a user dictating the order in which host names are assigned to multiple hosts, much less by inserting a disk or disks into the multiple hosts in a desired order.

For at least the reasons discussed above, *Paul* and *Klimenko*, whether considered alone or in combination, fail to teach or suggest several elements of Claim 1. For the same of analogous reasons, the *Paul* and *Klimenko* fail to teach or suggest similar elements of

independent Claim 16. Accordingly, the cited references cannot render Claims 1 or 16 obvious.

Therefore, Appellants contend that the arguments provided in the Final Office Action and maintained by the Advisory Action are clearly flawed and the teachings of the cited references do not render Claims 1 or 16 obvious. In addition, Appellants contend that the rejections of Claims 2-4 and 6 (which depend from Claim 1) and Claims 17-18 and 20 (which depend from Claim 16) are improper because such Claims depend from a claim shown to be allowable above.

**(2) Claims 9-11 and 22**

*Summary:*

*The rejection of independent Claim 9 as being unpatentable under 35 U.S.C. § 103(a) over* Paul *in view of* Klimenko *is improper because neither of* Paul *nor* Klimenko, *individually or in combination, disclose, teach or suggest the combination of elements recited in Claim 9. The rejection of dependent Claims 10-11 and 22 is improper at least because they depend from and provide further patentable elements to independent Claim 9.*

Claim 9 recites:

> 9. A program product for automatically naming hosts in a distributed data processing system, the program product comprising:
>
> (a) computer instructions enabling a controller in said distributed data processing system to:
>
> (i) receive a unique identifier (UID) from a first host in communication with a cluster controller, at least one of the plurality of hosts not having a fully functional operating system present thereon;
>
> (ii) in response to receiving the UID, cause the first host to produce a ready signal;
>
> (iii) receive user input from the first host, the user input;
>
> (iv) in response to receiving the user input from the first host, associate a first host name with the UID for the first host; and
>
> (v) after associating the first host name with the UID for the first host, cause the first host to produce a completion signal; and
>
> (b) a computer-usable medium encoding the computer instructions.

[Note: reference letters (a), (b), and (i)-(v) are included for reference and are not actually present in the pending claims.]

Appellants first wish to point out a clerical error in the text of independent Claim 9. In particular, element (a)(iii) of Claim 9 after the preamble recites "receive user input from the first host, the user input;" Element (a)(iii) should instead read "receive user input from the first host;" The clerical error occurred in a Response to Office Action dated August 14, 2006, in which element (a)(iii) was amended to remove particular text, but mistakenly did not remove the phrase ", the user input."

Appellants regret failing to notice this clerical error earlier. However, the clerical error was presumably not relevant to the Examiner's rejection of Claim 9, and if identified by the Examiner, would likely have merely resulted in a rejection under 35 U.S.C. § 112, second paragraph. Upon a favorable result to this Appeal, Appellants intend to file a simple amendment under 35 U.S.C. § 312 (a so-called "Section 312 Amendment") to amend element (a)(iii) as follows: "receive user input from the first host~~, the user input~~;"

Regarding the substantive rejection of Claim 9, Appellants contend that the Examiner's rejection of Claim 9 under 35 U.S.C. § 103(a) as being unpatentable over *Paul* in view of *Klimenko* is improper because neither of *Paul* nor *Klimenko*, individually or in combination, disclose, teach or suggest the combination of elements recited in Claim 9.

The Examiner rejected Claim 9 based on the same rationale as Claim 1. (*See* Final Office Action, Page 6: "Claims 9-11, 16-18 and 22 are directed to the same invention as claims 1-4 and 6. Therefore, the supporting rationale of the rejection to claims 1-4 and 6 applies equally as well to claims 9-11, 16-18 and 22").

Appellants contend that *Paul* and *Klimenko* fail to teach or suggest various elements of Claim 9 for similar or analogous reasons that *Paul* nor *Klimenko* fail to teach or suggest particular elements of Claim 9. For example:

- *Paul* and *Klimenko* fail to teach or suggest "receive a unique identifier (UID) from a first host in communication with a cluster controller, at least one of the plurality of hosts not having a fully functional operating system present thereon" for similar or analogous reasons that *Paul* nor *Klimenko* fail to teach or suggest element (a) of Claim 1, discussed above.

- *Paul* and *Klimenko* fail to teach or suggest "in response to receiving the UID, cause the first host to produce a ready signal" for similar or analogous reasons that *Paul* nor *Klimenko* fail to teach or suggest element (b) of Claim 1, discussed above.
- *Paul* and *Klimenko* fail to teach or suggest "in response to receiving the user input from the first host, associate a first host name with the UID for the first host" for similar or analogous reasons that *Paul* nor *Klimenko* fail to teach or suggest element (d) of Claim 1, discussed above.
- *Paul* and *Klimenko* fail to teach or suggest "after associating the first host name with the UID for the first host, cause the first host to produce a completion signal" for similar or analogous reasons that *Paul* nor *Klimenko* fail to teach or suggest element (e) of Claim 1, discussed above.

For at least the reasons discussed above, *Paul* and *Klimenko*, whether considered alone or in combination, fail to teach or suggest several elements of Claim 9. Accordingly, the cited references cannot render Claim 9 obvious.

Therefore, Appellants contend that the arguments provided in the Final Office Action and maintained by the Advisory Action are clearly flawed and the teachings of the cited references do not render Claim 9 obvious. In addition, Appellants contend that the rejections of Claims 10-11 and 22 (which depend from Claim 9) are improper because such Claims depend from Claim 9, shown to be allowable above.

***B. Rejection under 35 U.S.C. § 103(a) over* Paul *in view of* Klimenko *and further in view of* Park.**

**(1) Claims 5, 7, 8, 12, 14, 15, 19, and 21**

*Summary:*

*The rejection of dependent Claims 5, 7, 8, 12, 14, 15, 19, and 21 as being unpatentable under 35 U.S.C. § 103(a) over* Paul *in view of* Klimenko *and further in view of* Park *is improper because Claims 5, 7, 8, 12, 14, 15, 19, and 21 depend from and provide further patentable elements to independent Claims 1, 9, and 16 shown to be allowable in subsection A above.*

Appellants contend that the Examiner's rejection of dependent Claims 5, 7, and 8 as being unpatentable under 35 U.S.C. § 103(a) over *Paul* in view of *Klimenko* and further in view of *Park* is improper because Claims 5, 7, and 8 depend from and provide further patentable elements to independent Claim 1, shown to be allowable in Section VII(A)(1) above.

Appellants further contend that the Examiner's rejection of dependent Claims 19 and 21 as being unpatentable under 35 U.S.C. § 103(a) over *Paul* in view of *Klimenko* and further in view of *Park* is improper because Claims 19 and 21 depend from and provide further patentable elements to independent Claim 16, shown to be allowable in Section VII(A)(1) above.

Appellants further contend that the Examiner's rejection of dependent Claims 12, 14, and 15 as being unpatentable under 35 U.S.C. § 103(a) over *Paul* in view of *Klimenko* and further in view of *Park* is improper because Claims 12, 14, and 15 depend from and provide further patentable elements to independent Claim 9, shown to be allowable in Section VII(A)(2) above.

## SUMMARY

Appellants authorize the Commissioner to charge $510.00 for the Appeal Brief to Deposit Account No. 50-2148 of Baker Botts L.L.P.

Appellants also enclose a Petition for Extension of Time for two months and authorize the Commissioner to charge the amount of $460.00 to Deposit Account No. 50-2148.

Appellants believe there are no additional fees due at this time. However, the Commissioner is hereby authorized to charge any fees necessary or credit any overpayment to Deposit Account No. 50-2148 of Baker Botts L.L.P.

If there are any matters concerning this Application that may be cleared up in a telephone conversation, please contact Appellants' attorney at 512.322.2689.

Respectfully submitted,
BAKER BOTTS L.L.P.
Attorney for Appellants

Eric M. Grabski
Reg. No. 51,749

Date: December 13, 2007

CORRESPONDENCE ADDRESS:
CUSTOMER NO. **23640**
(512)322-2684
(512)322-8383 (fax)

## APPENDIX A - CLAIMS INVOLVED IN APPEAL

1. (Previously Presented) A method for automatically naming hosts in a distributed data processing system, the method comprising:

receiving a unique identifier (UID) at a cluster controller from each of a plurality of hosts in communication with the cluster controller, while at least one of the plurality of hosts is executing in a pre boot execution environment;

in response to receiving the UIDs, causing the plurality of hosts to produce ready signals;

receiving user input from a first host among the plurality of hosts, the user input comprising notification of the insertion of a disk within the first host;

in response to receiving the user input from a first host, associating a first host name with the UID for the first host;

after associating the first host name with the UID for the first host, causing the first host to produce a completion signal;

receiving user input from a second host among the plurality of hosts; and

repeating the operations of receiving replies from hosts, associating host names with UIDs, and causing hosts to produce completion signals, until each of the plurality of hosts has been named, such that the user input dictates the order in which host names are assigned to the multiple hosts.

2. (Original) The method of Claim 1, wherein the operation of associating a first host name with the UID for the first host comprises:

in response to receiving the user input from the first host, transmitting data to the first host; and

after transmitting the data to the first host, receiving a reply from the first host, such that the first host name is associated with the UID for the first host in further response to the reply.

3. (Original) The method of Claim 2, further comprising:

providing the cluster controller with a host-name index, wherein:

the operation of transmitting data to the first host comprises transmitting the host-name index to the first host;

the operation of receiving a reply from the first host comprises receiving an incremented host-name index from the first host; and

the operation of associating a host name with the UID for the first host comprises using the host-name index to generate the host name to be associated with the UID for the first host.

4. (Original) The method of Claim 2, further comprising:

providing the cluster controller with a host-name index and a host-name root; and

providing the multiple hosts with auto-naming logic, wherein:

the auto-naming logic causes the multiple hosts to transmit the UIDs to the cluster controller;

the auto-naming logic receives the index in the data from the cluster controller, increments the index, and transmits the incremented index to the cluster controller in the reply; and

the operation of associating a host name with the UID for the first host comprises using the host-name root and the host-name index to generate the host name to be associated with the UID for the first host.

5. (Original) The method of Claim 1, wherein the operation of causing the multiple hosts to produce ready signals comprises activating light emitting diodes (LEDs) on the multiple hosts to indicate that the multiple hosts are ready to be named.

6. (Previously Presented) The method of Claim 1, wherein the operation of receiving user input from the first host comprises detecting that a blank disk has been inserted into a disk drive of the first host.

7. (Original) The method of Claim 1, wherein the operation of causing the first host to produce a completion signal comprises deactivating a light emitting diode (LED) on the first host.

8. (Original) The method of Claim 1, wherein the operation of causing the first host to produce a completion signal comprises producing an audible signal to indicate that the first host has been named.

9. (Previously Presented) A program product for automatically naming hosts in a distributed data processing system, the program product comprising:

computer instructions enabling a controller in said distributed data processing system to:

receive a unique identifier (UID) from a first host in communication with a cluster controller, at least one of the plurality of hosts not having a fully functional operating system present thereon;

in response to receiving the UID, cause the first host to produce a ready signal;

receive user input from the first host, the user input;

in response to receiving the user input from the first host, associate a first host name with the UID for the first host; and

after associating the first host name with the UID for the first host, cause the first host to produce a completion signal; and

a computer-usable medium encoding the computer instructions.

10. (Previously Presented) The program product of Claim 9, wherein:

the computer instructions respond to the user input from the first host by transmitting data to the first host;

the computer instructions receive a reply from the first host; and

the computer instructions associate the first host name with the UID for the first host in further response to the reply.

11. (Original) The program product of Claim 10, wherein the operations performed by the computer instructions further comprise:

recognizing a host-name index; and

transmitting the host-name index to the first host with the data, wherein:

the operation of receiving a reply from the first host comprises receiving an incremented host-name index from the first host; and

the operation of associating a host name with the UID for the first host comprises using the host-name index to generate the host name to be associated with the UID for the first host.

12. (Previously Presented) The program product of Claim 9, wherein the computer instructions cause the first host to produce a ready signal by activating a light emitting diode (LED) each of the respective plurality of hosts to indicate that the multiple hosts are ready to be named.

13. (Cancelled)

14. (Original) The program product of Claim 9, wherein the computer instructions cause the first host to produce a completion signal by deactivating a light emitting diode (LED) on the first host.

15. (Original) The program product of Claim 9, wherein the computer instructions cause the first host to produce a completion signal by producing an audible signal to indicate that the first host has been named.

16. (Previously Presented) A data processing system for automatically naming hosts in a distributed data processing system, the data processing system comprising:

a network interface in communication with a plurality of hosts, a processor in communication with the network interface, data storage in communication with the processor, and computer instructions stored in the data storage, wherein, when the computer instructions are executed by the processing resources, the computer instructions perform operations comprising:

receiving a unique identifier (UID) from each of a plurality of the plurality of hosts;

in response to receiving the UIDs, causing the plurality of hosts to produce ready signals;

receiving user input from a first host among the multiple hosts, the user input comprising a signal indicative of an insertion of a disk within a disk drive of the first host;

in response to receiving the user input from the first host, associating a first host name with the UID for the first host without regard to data, if any, stored on the disk;

after associating the first host name with the UID for the first host, causing the first host to produce a completion signal;

receiving user input from a second host among the plurality of hosts; and

repeating the operations of receiving replies from hosts, associating host names with UIDs, and causing hosts to produce completion signals, until each of the plurality of hosts has been named, such that the user input dictates the order in which host names are assigned to the plurality of hosts.

17. (Original) The data processing system of Claim 16, wherein the operation of associating a first host name with the UID for the first host comprises:

transmitting data to the first host; and

receiving a reply from the first host, wherein the computer instructions associate the first host name with the UID for the first host in further response to the reply.

18. (Original) The data processing system of Claim 17, wherein the operations performed by the computer instructions further comprise

recognizing a host-name index; and

transmitting the host-name index to the first host with the data, wherein:

the operation of receiving a reply from the first host comprises receiving an incremented host-name index from the first host; and

the operation of associating a host name with the UID for the first host comprises using the host-name index to generate the host name to be associated with the UID for the first host.

19. (Previously Presented) The data processing system of Claim 16, wherein the computer instructions cause the plurality of hosts to each produce a ready signal by activating a light emitting diode (LED) on each of the plurality of hosts to indicate that each of the plurality of hosts is ready to be named.

20. (Previously Presented) The data processing system of Claim 16, wherein the user input comprises signals indicating that a blank disk has been inserted into a disk drive of the first host.

21. (Original) The data processing system of Claim 16, wherein the computer instructions cause the first host to produce a completion signal by deactivating a light emitting diode (LED) on the first host.

22. (Original) The program product of claim 9, wherein said user input from the first host comprises a signal indicative of insertion of a disk into a disk drive of the system and where said associating a first host name with the UID for the first host comprises associating said first host name without regard to data, if any, stored on the disk.

## APPENDIX B - EVIDENCE

**NONE**

## APPENDIX C: RELATED PROCEEDINGS

**NONE**